

# 무선 센서 네트워크를 위한 향상된 센서 MAC 프로토콜

정회원 이주아\*, 김재현\*, 민승욱\*\*

## Improved Sensor MAC Protocol for Wireless Sensor Network

Ju-a Lee\*, Jae-hyun Kim\*, Seung-wook Min\*\* *Regular Members*

### 요 약

센서 네트워크에서는 배터리 용량이 중요한 문제이다. 기존 MAC 프로토콜에서는 배터리 수명을 연장시키기 위하여 sleep 모드와 active 모드가 주기적으로 전환되는 S-MAC 프로토콜과 가변적인 active 시간을 사용하는 T-MAC 프로토콜이 제안되었다. 본 논문에서는 S-MAC 프로토콜과 T-MAC 프로토콜보다 에너지 소비 효율성이 뛰어난 IS-MAC (Improved Sensor MAC) 프로토콜을 제안한다. IS-MAC 프로토콜은 센서 네트워크의 에너지 효율성을 높이기 위하여 버퍼의 임계 값을 사용하여 데이터를 전송하며 에너지 낭비의 원인인 제어 패킷 수를 줄이는 방안을 제안하였다. 또한 제안한 알고리즘에 대한 수학적 분석을 통하여 IS-MAC이 기존의 센서 MAC 프로토콜보다 에너지와 데이터 전송 시간 면에서 더 효율적임을 보였다.

**Key Words** : IS-MAC, Sensor Network, energy efficiency, transmission delay, MAC protocol

### ABSTRACT

It is very important for the sensor network to save battery capacity. Switching active mode to sleep mode is used for S-MAC protocol and expiring timer before end of the active part is used for T-MAC in order to extend battery life span. We proposed IS-MAC (Improved Sensor MAC) which gives more energy efficiency than S-MAC and T-MAC. To improve energy efficiency in sensor network, we used the threshold value in buffer to transmit data packet and proposed the method to reduce the number of control packets which cause extra battery consumption. Based on the analytical results, we found that the proposed IS-MAC protocol shows better performance than conventional MAC protocols.

### I. 서론

센서 네트워크의 통신은 여러 계층으로 나누어지고 그 중의 MAC (Medium Access control) 계층은 네트워크가 성공적으로 동작할 수 있게 해주는 중요한 기능을 한다. 그리고 MAC 프로토콜은 가까이 있는 두 노드가 동시에 데이터를 전송하여 충돌이 일어나는 것을 피할 수 있게 해준다. 무선 센서 네

트워크를 위한 MAC 프로토콜은 S-MAC (Sensor-MAC) 프로토콜<sup>[1]</sup>과 T-MAC (Timeout- MAC) 프로토콜<sup>[2]</sup> 등 여러 가지가 있으며 센서 네트워크는 기존의 네트워크와 달리 크기와 배터리 용량이 작기 때문에 QoS (Quality of Service) 요구보다는 에너지 소비를 최소화하는 것이 우선시된다<sup>[1,2]</sup>. MAC 프로토콜에서 에너지 낭비는 idle listening, 충돌, 프로토콜 오버헤드, overhearing 때문에 일어난다<sup>[2]</sup>.

※ 본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅및네트워크원천기술개발사업과 2006년도 2단계 BK21 사업에 의하여 지원되었습니다.

\* 아주대학교 전자공학과 무선인터넷 연구실 (gaia@ajou.ac.kr, jkim@ajou.ac.kr) \*\* 한국전자통신연구원 (swmin@etri.re.kr)

논문번호 : KICS2006-02-082, 접수일자 : 2006년 2월 15일, 최종논문접수일자 : 2006년 11월 5일

이 중에서도 idle listening은 에너지 낭비의 제일 큰 요인이다.

IEEE 802.11 ad hoc 모드 또는 CSMA (Carrier Sense Multiple Access) 기반의 MAC 프로토콜을 사용하는 노드들은 트래픽을 받기 위해 항상 채널을 감지하여야 한다. 이는 실제 트래픽이 전송되지 않는 대기 (idle) 상태에서 에너지를 낭비하는 것을 의미한다. Mark의 시뮬레이션 결과와 IEEE 802.11 표준에는 대기 모드, 수신 모드, 송신 모드 사이의 에너지 소비 비율이 각각 1:1.05:1.4와 1:2:2.5 라고 나온다<sup>[3],[4]</sup>. 이를 해결하기 위한 프로토콜 중 하나인 S-MAC 프로토콜에서는 시간을 active 구간과 sleep 구간으로 구성된 프레임으로 나눈다. sleep 구간에는 다른 노드와 무선으로 신호를 주고받는 transceiver의 전원을 차단하여 데이터를 전송하지 않고 버퍼에 저장한다. active 구간에서는 저장된 데이터를 이웃 노드로 전송한다. 그러나 S-MAC 프로토콜의 duty cycle은 높은 부하를 기준으로 고정되었기 때문에 트래픽이 변화하는 환경에서는 적절하지 않다. S-MAC 프로토콜을 발전시킨 T-MAC 프로토콜에서는 active 모드에서도 데이터 송수신이 없다는 것을 감지하면 타이머를 작동시켜 특정 시간 이후에 sleep 모드가 된다. 이러한 타이머는 프로토콜의 에너지 효율을 향상시킨다. [5]에서 제안한 E<sup>2</sup>-MAC (Energy Efficiency-MAC) 프로토콜은 본 논문과 유사하나 E<sup>2</sup>-MAC 프로토콜의 타이머는 홉 수에 따라 변하기 때문에 트래픽이 변화하는 환경에서는 적절하지 못하다. 따라서 본 논문에서는 S-MAC과 T-MAC 프로토콜의 장점들을 사용하고 타이머 시간을 다시 정의하여 에너지 효율을 높이는 IS-MAC (Improved Sensor MAC) 프로토콜을 제안한다. 또한 노드별 공평성, 채널 용량 활용과 같은 QoS를 고려한다. 2장에서는 IS-MAC 프로토콜을 소개하고, 3장에서 수학적 분석 및 결과를 설명하며 이에 대한 분석 결과를 4장에서 살펴보고, 5장에서 결론을 맺는다.

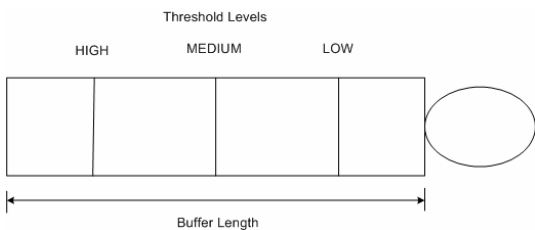


그림 1. 임계 값을 사용하는 버퍼

## II. IS-MAC 프로토콜

제안한 IS-MAC 프로토콜은 그림 1과 같이 센서 노드 내의 버퍼의 임계 값을 HIGH, MEDIUM, LOW로 지정한다. 이 임계 값은 버퍼 길이에 따라 달라지고 저장된 데이터가 지정된 임계 값을 초과하면 데이터가 전송된다. 반면 타이머는 T-MAC 프로토콜보다 더 작게 설정해준다. 노드는 타이머가 동작하는 동안 RTS (Request-to-Send), CTS (Clear-to-Send), FRTS (Future RTS) 제어 패킷을 수신 받거나 어떠한 데이터 패킷도 송신하지 않으면 sleep 모드가 된다. 기존의 프로토콜은 트래픽 양이 다양하게 변화하면 효율성이 감소하는 반면에 제안한 프로토콜은 임계 값을 사용하여 에너지 효율을 최적화하고, 세 단계의 임계 값을 사용함으로써 노드별 공평성을 증가시킨다.

데이터 전송 기준이 되는 임계 값에 대하여 살펴보면  $Th_l, Th_m, Th_h$ 는 LOW, MEDIUM, HIGH의 임계 값을 나타낸다.  $q_{le}$ 는 버퍼 크기이고, 각 임계 값에 대한 가중치는  $a_l, a_m, a_h$ 로  $0 < a_l \leq 0.25, 0 < a_m \leq 0.5, 0 < a_h < 1$ 로 설정하였다. 각 가중치의 범위는 정해진 것이 아니며 변경 가능하다. 임계 값은 식 (1)과 같이 표현할 수 있다.

$$Th_k = a_k \times q_{le}, \quad k = l, m, h \quad (1)$$

본 논문에서 고려한 네트워크의 구조는 그림 2와 같이 선형 토폴로지로 되어있고, 모든 노드는 자신의 위치를 알고 있다고 가정한다. 각 원은 원 중심에 위치한 노드의 전송 범위를 나타낸다.

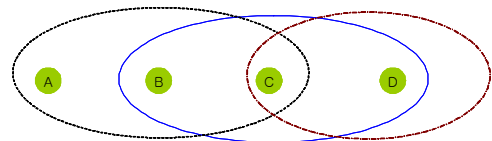


그림 2. 선형 토폴로지

### 2.1 IS-MAC 프로토콜의 동작

무선 네트워크에서는 hidden node problem과 exposed node problem을 해결하기 위하여 데이터 전송 전에 RTS/CTS 패킷을 송수신한다. 그러나 하나의 데이터를 여러 홉 동안 전송해야 하는 경우 그만큼 RTS/CTS 패킷의 전송 부하가 증가한다. IS-MAC 프로토콜에서는 RTS/CTS 패킷 전송 부하를 줄이기 위하여 FRTS 패킷을 사용한다.

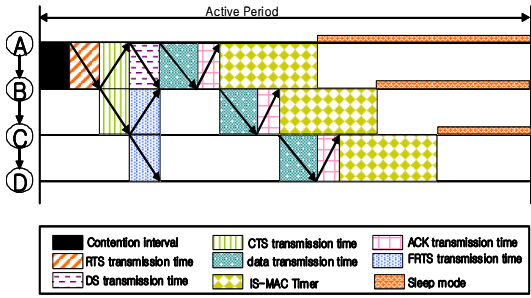


그림 3. IS-MAC 프로토콜

그림 3은 A 노드에서 D 노드 방향으로 있는 선형 토폴로지의 경우 IS-MAC 프로토콜의 동작을 나타내고 있으며 자세한 동작은 다음과 같다.

- A 노드의 버퍼 크기( $Q_{size}$ )가 임계 값 이하 ( $Q_{size} < Th_t$ )이면 노드는 타이머를 작동시킨다. 타이머의 작동이 끝날 때까지 주위 노드로부터 RTS, CTS, FRTS 패킷을 전송 받지 못한다면 sleep 모드가 된다.
- A 노드의 버퍼 크기가 임계 값 이상 ( $Q_{size} \geq Th_t$ ) 이면 RTS 패킷을 B 노드에게 전송한다.
- RTS 패킷을 수신한 B 노드는 CTS 패킷을 브로드캐스트하여, A 노드와 C 노드에게 전송한다.
- CTS 패킷을 전송 받은 C 노드는 D 노드에게 FRTS 패킷을 전송하여 채널을 예약한다. FRTS 패킷을 수신한 D 노드는 C 노드 이외의 다른 노드와 데이터를 송수신할 수 없다.
- C 노드의 전송은 B 노드까지 영향을 미치므로 C 노드가 FRTS 패킷을 D 노드에게 전송할 때 A 노드가 B 노드에게 데이터 패킷을 전송하면 B 노드에서 충돌이 생긴다. 이를 방지하기 위하여 C 노드가 FRTS 패킷을 D 노드에게 전송하는 동안 A 노드는 B 노드에게 DS (Data-Send) 패킷을 전송한다. DS 패킷에는 특별한 정보가 실리지 않으므로 충돌이 일어나도 문제가 되지 않는다.
- A 노드는 DS 패킷 전송 이후에 B 노드로 데이터 패킷을 전송한다.
- A 노드가 B 노드에게 데이터를 전송하고 있는 동안 D 노드는 그 다음 노드에게 FRTS 패킷을 전송하여 채널을 예약할 수 있으나 본 논문에서는 채널 효율성을 위해 채널 예약을 D 노드까지로 제한하였다.

- 데이터를 수신한 B 노드는 A 노드에게 ACK 패킷을 전송한 뒤 별도의 RTS/CTS 패킷 교환 없이 바로 C 노드에게 데이터를 전송한다.
- 데이터를 수신한 C 노드는 B 노드와 동일한 방법으로 D 노드에게 데이터를 전송한다.

IS-MAC에서는 FRTS 패킷을 이용하여 채널을 예약함으로써 에너지 효율과 함께 전송 지연을 크게 향상시킨다. 그러나 FRTS 패킷을 수신한 노드들은 통신에 참여하지 않은 다른 노드와는 데이터를 송수신 할 수 없기 때문에 채널 사용 효율성은 떨어진다. 그러므로 FRTS 패킷이 전송될 수 있는 홉 수는 적절하게 제한되어야 한다. 본 논문에서는 채널 효율성을 고려하여 FRTS 패킷으로 예약 가능한 홉 수를 1홉으로 제한하였다. 즉, 1개의 RTS 패킷으로 전송 가능한 홉 수는 3홉까지로 제한된다.

### 2.2 Contention Window

IS-MAC 프로토콜에서 각 노드는 HIGH, MEDIUM, LOW의 다른 우선 순위를 가지며 contention window는 이러한 우선 순위에 따라 달라지므로 노드별 공평성을 높여준다.

IS-MAC 프로토콜의 contention window를 구하기 위하여 각 노드의 contention window를  $t_{IS-cont}$  라 하고  $r$ 을 재전송 횟수라 정의한다.  $p_{bt}$ 는 각 노드의 우선 순위이며  $p_{bt} = 1 - a$ 로  $a$ 는  $\{a_i, a_m, a_h\}$ 의 원소이다.  $p_{bt}$ 는 값이 작을수록 우선 순위가 높아지는 것으로 즉, 버퍼에 있는 데이터가 많을수록 우선 순위가 높아진다. contention window는 식 (2)와 같이 정의되며  $p_{bt}$ 와  $r$ 의 함수이다.

$$t_{IS-cont} = \lfloor p_{bt} \times CW \times RAN() \rfloor \times slottime \quad (2)$$

재전송된 횟수에 따라 contention window가 지수적으로 증가할 수 있도록  $CW = CW_{min} \times 2^r$ 으로 정의한다. 여기서  $CW_{min}$ 은 contention window의 최소 값이다.  $RAN()$ 은 0과 1사이의 값이고  $\lfloor * \rfloor$ 은 \*보다 작거나 같은 수 중 제일 큰 정수이다.

### 2.3 타이머

T-MAC 프로토콜에서 노드가 sleep 모드가 되기 위한 타이머 조건은 식 (3)과 같다<sup>2)</sup>. T-MAC 프로토콜의 타이머는 노드가 데이터를 송수신하지 않는 상태에서의 불필요한 에너지 소비를 줄여준다.

$$TO_{T-mac} > T_{cont} + T_{RTS} + T_{SIFS} \quad (3)$$

이때,  $T_{cont}$ 는 contention interval 시간이고,  $T_{RTS}$ 는 RTS 패킷 전송 시간,  $T_{SIFS}$ 는 Short Inter Frame Space이다. 이와 유사하게 IS-MAC 프로토콜에서 노드가 sleep 모드로 전환되기 위한 타이머는 식 (4)와 같이 정의할 수 있다.

$$TO_{IS-mac} > T_{RTS} + T_{CTS} + T_{FRTS} \quad (4)$$

이때,  $T_{FRTS}$ 는 FRTS 패킷 전송 시간이다.

IS-MAC 프로토콜은 같은 프레임 내에서 데이터 패킷을 3 홉까지 전송할 수 있다. IS-MAC 프로토콜에서 타이머는 contention interval 시간이 끝난 후 동작하고 T-MAC 프로토콜의 타이머보다 짧기 때문에 idle listening으로 인한 에너지 낭비를 감소시킬 수 있다.

### III. 수학적 분석

본 논문에서는 기존의 MAC 프로토콜과 IS-MAC 프로토콜의 성능을 수학적으로 분석해 비교하였다.

#### 수식을 위한 파라미터 정의

$t_{A, total}$	총 프레임 동안의 active 시간
$t_{A, m}$	$m$ 번째 프레임의 active 시간
$t_{A, p=mac}$	T-MAC 프로토콜에서 한 프레임 동안에 전송 패킷이 없는 경우 active 시간
$t_{A, q=mac}$	T-MAC 프로토콜에서 한 프레임 동안에 전송 패킷이 있는 경우 active 시간
$t_{cont, n}$	$n$ 번째 홉에서의 평균 contention window
$t_{IS-cont, n}$	IS-MAC 프로토콜에서 $n$ 번째 홉의 contention window
$t_{on, m}$	$m$ 번째 프레임의 active 구간이 시작하는 시간
$t_{off, m}$	$m$ 번째 프레임의 active 구간이 끝나는 시간
$t_{l, m}$	$m$ 번째 프레임의 active 구간에서 패킷 전송 시간
$TR_c$	제어 패킷 전송 시간
$TR_{data}$	데이터 패킷 전송 시간
$TR_{ack}$	응답 패킷 전송 시간
$TO_{T-mac}$	T-MAC 프로토콜의 타이머
$TO_{IS-mac}$	IS-MAC 프로토콜의 타이머
$E_{tx}$	단위 시간당 패킷 전송에 소비되는 에너지
$E_{rx}$	단위 시간당 패킷 수신에 소비되는 에너지

$E_{idle}$	단위 시간당 idle listening 동안 소비되는 에너지
$p_{T-mac}$	T-MAC 프로토콜에서 active 시간 동안 데이터를 전송하지 않을 확률
$q_{T-mac}$	T-MAC 프로토콜에서 active 시간 동안 데이터를 전송할 확률
$P_{IS-no}$	IS-MAC 프로토콜에서 active 시간 동안 데이터를 전송하지 않을 확률
$q_{IS-mac}$	IS-MAC 프로토콜에서 active 시간 동안 데이터를 전송할 확률
$N_{ideal}$	S-MAC 프로토콜과 T-MAC 프로토콜에서 한 개의 RTS/CTS 패킷으로 예약 가능한 홉 수 (2 홉)
$N_{IS-mac}$	IS-MAC 프로토콜에서 한 개의 RTS/CTS 패킷으로 예약 가능한 홉 수 (3 홉)
$E_{ch}$	IS-MAC 프로토콜의 채널 사용률

#### 3.1 데이터 전송 지연

본 절에서는 각 MAC 프로토콜에서의 전송 지연 시간을 분석하여 비교해보았다. 전송 지연을 분석하기에 앞서 각 지연 요소를 살펴보면 우선 경쟁을 위한 contention window와 재전송으로 인한 back-off 지연, 전송을 위한 transmission 지연, propagation 지연, processing 지연, 패킷이 버퍼에 있는 시간인 큐잉 지연을 생각해 볼 수 있다<sup>[1]</sup>. 본 논문에서는 트래픽 양이 적고, 1 홉 사이에 전송되는 데이터 패킷을 한 개라고 가정하여, propagation 지연과 processing 지연, 큐잉 지연, back-off 지연을 무시한다. 그리고 프레임의 시작을 active 구간이라고 가정한다.

S-MAC 프로토콜과 T-MAC 프로토콜에서 한 프레임 동안 1 개의 데이터 패킷을 전송할 때 전송 가능한 최대 홉 수는 active 시간  $t_A$ 에 따라 다르며 식 (5)와 같다.

$$N = \left\lfloor \frac{t_A}{t_{cont, n} + 2TR_c + TR_{data} + TR_{ack}} \right\rfloor \quad (5)$$

그림 4는 active 구간에서의 패킷 전송을 보여주고 있으며 식 (5)를 표현 한 것이다. 1 홉 사이에서 데이터를 전송할 때는 한 번의 contention window와 RTS/CTS 패킷인 제어 패킷 전송 시간이 필요하고, 데이터와 ACK 패킷의 전송 시간이 필요하다. 그러므로 active 구간 동안 몇 홉을 전송할 수 있는냐는 1 홉 사이에서 데이터를 전송하기 위해 필요

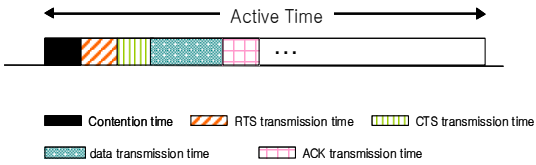


그림 4. Active 구간에서의 패킷 전송

한 시간이 몇 개가 들어가는가에 따라 결정된다.

그림 5는 S-MAC 프로토콜에서 3 홉 동안의 패킷 교환을 보여준다. T-MAC 프로토콜은 그림 5와 유사하며 단지 S-MAC 프로토콜과 다른 점은 타이머가 동작하여 active 구간에서 패킷 교환이 없을 경우, S-MAC 프로토콜보다 더 빨리 sleep 모드로 전환된다는 것이다.

데이터가 한 프레임 동안  $N$ 홉 만큼 전송될 수 있다면 S-MAC과 T-MAC 프로토콜에서의 데이터 전송 지연  $D_{S-mac}$  은 식 (6)과 같다.

$$\begin{aligned}
 D_{S-mac} &= \sum_{n=1}^N D_n \\
 &= \sum_{n=1}^N (t_{cont, n} + N(2TR_c + TR_{tx} + TR_{ack}))
 \end{aligned} \tag{6}$$

반면에, IS-MAC 프로토콜에서 FRTS 패킷으로 예약 가능한 홉 수를 1 홉이라고 했을 때 IS-MAC 프로토콜에서 데이터 전송 지연  $D_{IS-mac}$  은 식 (7)로 주어진다.

$$\begin{aligned}
 D_{IS-mac} &= \sum_{n=1}^N (k_n t_{IS-cont, n} + 3k_n TR_c + TR_{tx} + TR_{ack}) \\
 k_n &= \begin{cases} 1, & n=1, 4, 7, 10, \dots \\ 0, & otherwise \end{cases}
 \end{aligned} \tag{7}$$

식 (7)은 앞 장의 그림 3을 보면 분명히 알 수 있다. 본 논문에서는 한 개의 RTS 패킷으로 전송 가능한 홉 수를 3 홉으로 제한하였고, 첫 번째 홉

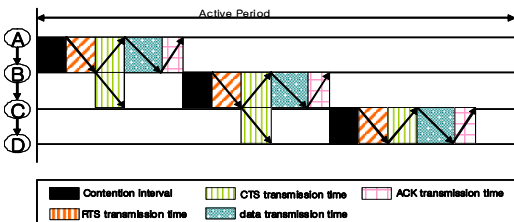


그림 5. S-MAC 프로토콜에서의 패킷 교환

을 제외한 나머지 홉에서는 RTS/CTS 패킷 교환 없이 바로 데이터를 전송하기 때문에 제어 패킷으로 인한 지연 시간은 3 홉 마다 나타난다.

### 3.2 제어 패킷 오버헤드

S-MAC과 T-MAC 프로토콜에서  $N$  홉 동안 데이터를 전송할 때 발생하는 제어 패킷 오버헤드 수 CPO (Control Packet Overhead)는  $2N$ 이다.

IS-MAC 프로토콜에서는 FRTS 패킷을 사용하여 한 개의 RTS 패킷으로 데이터를 3 홉 동안 전송할 수 있기 때문에 제어 패킷 오버헤드 수가 감소한다.  $N$  홉 동안 데이터를 전송할 때 발생하는 CPO는 식 (8)과 같다.

$$\begin{aligned}
 CPO &= \sum_{n=1}^N k_n, \\
 k_n &= \begin{cases} 3, & n=1, 4, 7, 10, \dots \\ 1, & n=2, 5, 8, 11, \dots \\ 0, & otherwise \end{cases}
 \end{aligned} \tag{8}$$

위 식을 다시 정리해보면 식 (9)와 같다.

$$CPO = 3 \left\lfloor \frac{N+2}{3} \right\rfloor + \left\lfloor \frac{N+1}{3} \right\rfloor \tag{9}$$

### 3.3 Energy Consumption

본 절에서는 S-MAC, T-MAC 및 IS-MAC 프로토콜의 에너지 소비를 비교하기 위하여 데이터를 수신 받기 전의 노드는 sleep 모드로 전환되지 않는다고 가정하였다.

S-MAC 프로토콜에서 데이터가 1 개의 프레임 구간 동안  $N$  홉만큼 전송될 때 사용되는 에너지  $E_{S-mac}$  은 식 (10)과 같다.

$$\begin{aligned}
 E_{S-mac}(N) &= N(2TR_c + TR_{data} + TR_{ack})(E_{tx} + E_{rx}) \\
 &\quad + (N-1)TR_c E_{rx} \\
 &\quad + \left[ \sum_{n=1}^N t_{cont, n} + 2(t_A - TR_{set1}) \right. \\
 &\quad \left. + (N-1)(t_A - 2TR_{set1} - TR_c) \right] E_{idle}
 \end{aligned} \tag{10}$$

이때,  $TR_{set1} = 2TR_c + TR_{tx} + TR_{ack}$  이다.

T-MAC 프로토콜에서 데이터가 1 개의 프레임 구간 동안  $N$  홉만큼 전송될 때 사용되는 에너지  $E_{T-mac}$  은 식 (11)과 같다.

$$\begin{aligned}
 E_{T-mac}(N) &= N(2TR_c + TR_{data} + TR_{ack})(E_{tx} + E_{rx}) \\
 &\quad + (N-1)TR_c E_{rx} \\
 &\quad + \left[ \sum_{k=1}^{N+1} \sum_{n=1}^k t_{cont, n} - t_{cont, N+1} \right. \\
 &\quad \quad + (N+1)TO_{T-mac} \\
 &\quad \quad + (N-1)(TR_{set1} - TR_c) \\
 &\quad \quad \left. + \frac{(N-1)(N-2)}{2} TR_{set1} \right] E_{idle}
 \end{aligned} \tag{11}$$

IS-MAC 프로토콜에서 고정된 active 구간 동안 데이터는 3 홉까지 전송될 수 있다.  $m$ 개의 프레임 구간 동안  $N$  홉만큼 전송될 때 사용되는 에너지  $E_{IS-mac}$  은 식 (12)와 같다.

$$\begin{aligned}
 E_{IS-mac}(N) &= \left( 3 \left\lfloor \frac{N+2}{3} \right\rfloor + \left\lfloor \frac{N+1}{3} \right\rfloor \right) TR_c E_{tx} \\
 &\quad + \left( 2 \left\lfloor \frac{N+2}{3} \right\rfloor + N \right) TR_c E_{rx} \\
 &\quad + N(TR_{data} + TR_{ack})(E_{tx} + E_{rx}) \\
 &\quad \quad + [NTO_{IS-mac} + (N+1)t_{cont, 1} \\
 &\quad \quad + \sum_{l=1}^{\lfloor \frac{N-1}{3} \rfloor} \sum_{n=3l+1}^N (n-3l+1)t_{cont, 3l+1} \\
 &\quad \quad + \sum_{n=1}^{N-1} \left\lfloor \frac{n}{3} \right\rfloor TR_{set2} + \left( 2 \left\lfloor \frac{N+1}{3} \right\rfloor + \left\lfloor \frac{N}{3} \right\rfloor \right) TR_c \\
 &\quad \quad + \sum_{n=1}^N \left( \left\lfloor \frac{n}{3} \right\rfloor + \left\lfloor \frac{n-1}{3} \right\rfloor \right) (TR_{data} + TR_{ack}) E_{idle}
 \end{aligned} \tag{12}$$

이때,  $TR_{set2} = 3TR_c + TR_{tx} + TR_{ack}$  이다.

### 3.4 Active 시간

센서 네트워크에서 에너지는 active 시간 동안 소비되므로 여러 MAC 프로토콜을 비교해 보기 위하여 active 시간을 분석한다.

S-MAC 프로토콜에서 전체 active 시간은 식 (13)과 같다.

$$t_{A, total} = \sum_{m=1}^M (t_{off, m} - t_{on, m}) \tag{13}$$

여기서,  $M$ 은 프레임 수이다.

T-MAC 프로토콜은 타이머를 두어 일정 시간 동안 패킷이 도착하지 않으면 sleep 모드로 전환이 된다. 한 프레임 동안 송수신 패킷이 없는 경우의 active 시간은 식 (14)와 같다.

$$t_{A, p_{T-mac}} = p_{T-mac} TO_{T-mac} \tag{14}$$

한 프레임 동안 송수신 패킷이 있는 경우의 active 시간은 식 (15)와 같다.

$$t_{A, q_{T-mac}} = q_{T-mac} (t_l + TO_{T-mac}) \tag{15}$$

그러므로 T-MAC 프로토콜에서 한 프레임 동안의 active 시간은 식 (16)과 같다. 이때,  $p_{T-mac} + q_{T-mac} = 1$ 이다.

$$\begin{aligned}
 t_{A, total} &= \sum_{m=1}^M (t_{A, p_{T-mac}} + t_{A, q_{T-mac}}) \\
 &= M p_{T-mac} TO_{T-mac} \\
 &\quad + q_{T-mac} \sum_{m=1}^M (t_{l, m} + TO_{T-mac})
 \end{aligned} \tag{16}$$

IS-MAC 프로토콜에서 active 시간은 T-MAC 프로토콜과 유사하게 식 (17)과 같다.

$$\begin{aligned}
 t_{A, total} &= M p_{IS-mac} TO_{IS-mac} \\
 &\quad + q_{IS-mac} \sum_{m=1}^M (t_{l, m} + TO_{IS-mac})
 \end{aligned} \tag{17}$$

그러나 IS-MAC 프로토콜은 T-MAC 프로토콜보다 전송 시간과 타이머 시간이 짧고,  $q_{T-mac}$  이  $q_{IS-mac}$  보다 작기 때문에 active 시간에서 차이가 난다. T-MAC 프로토콜에서 노드가 전송에 참여하지 않을 때( $Q_{size} = 0$ )는 다른 노드로부터 패킷을 수신하지 못하는 경우이다. IS-MAC 프로토콜에서 노드가 전송에 참여하지 않을 때( $0 \leq Q_{size} \leq Th_i$ )는 다른 노드로부터 패킷을 전송 받지 않을 때이다. 그러므로  $q_{T-mac}$  이  $q_{IS-mac}$  보다 작은 값을 갖는다. 세 프로토콜을 비교해보면 S-MAC 프로토콜은 정해진 active 시간이 있으므로 프레임 수에 따라 active 시간이 배수로 증가한다. T-MAC과 IS-MAC 프로토콜은 타이머가 있어 active 시간이 데이터 전송 량에 따라 달라진다. T-MAC과 IS-MAC 프로토콜의 active 시간을 구하는 식은 유사하나 T-MAC 프로토콜의 타이머 값과 데이터 전송 시간이 IS-MAC 프로토콜보다 더 길고 데이터를 전송하지 않을 확률이 더 작아서 T-MAC의 active 시간이 IS-MAC의 active 시간보다 더 길다.

### 3.5 채널 효율성

기존의 방법에서는 한 노드가 자신이 받은 데이터를 다음 노드로 전송할 때 RTS/CTS 메시지 때문에 데이터를 바로 전송할 수 없었다. 그러나 FRTS 방법은 한 개의 RTS/CTS 패킷으로 연속적

인 통신을 가능하게 한다. FRTS로 예약된 노드들 즉, FRTS를 받은 노드들은 자신의 데이터를 전송할 수 없는 대기 상태가 되기 때문에 채널 사용률이 떨어진다. 본 논문에서는 식 (18)로 정의한 채널 사용률에 의하여 최대 3 홉까지 예약할 수 있다고 가정하였다.

$$E_{CH} = \left( \frac{N_{ideal}}{N_{IS-mac}} \right) \times 100 \quad (18)$$

#### IV. 분석 결과

본 논문에서는 데이터 전송 지연, 제어 패킷 수, 전송에 필요한 에너지를 수식적으로 분석하여 IS-MAC 프로토콜이 S-MAC과 T-MAC 프로토콜보다 에너지 효율성이 뛰어나다는 것을 보였다.

그림 6은 S-MAC, T-MAC 및 IS-MAC 프로토콜에서 데이터 전송 홉 수에 대한 전송 지연 시간을 나타낸 것이다. 그림 6에서 보듯이 1 개의 데이터 패킷을 6 홉 동안 전송 할 때에는 IS-MAC 프로토콜의 데이터 전송 시간이 S-MAC과 T-MAC 프로토콜보다 약 55% 감소하였다. IS-MAC 프로토콜이 다른 MAC 프로토콜보다 전송 시간이 감소하는 이유는 S-MAC과 T-MAC 프로토콜에 비하여 RTS/CTS 패킷 전송 시간이 감소하기 때문이다.

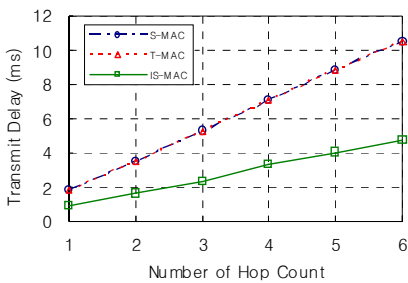


그림 6. 데이터 전송 지연

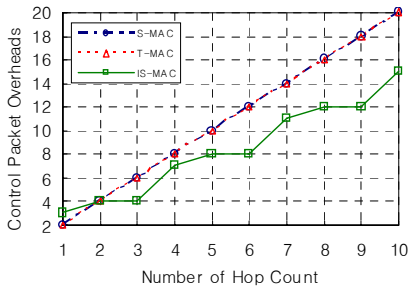


그림 7. 제어 패킷 수

그림 7은 각 MAC 프로토콜의 데이터 전송 홉 수에 따른 제어 패킷 수를 나타낸 것으로 그림 7에서 알 수 있듯이 데이터 전송 홉 수가 증가함에 따라 다른 MAC 프로토콜과 IS-MAC 프로토콜의 제어 패킷 수 차이가 증가한다. 제어 패킷 전송은 센서 네트워크에서 주요 에너지 소비 요인 중 하나이기 때문에, 전송되는 제어 패킷 수가 감소할수록 소비되는 에너지도 감소한다. 결국, IS-MAC 프로토콜에서 데이터 전송은 전송되는 홉 수가 증가할수록 다른 MAC 프로토콜보다 에너지 소비가 적어진다 것을 의미한다.

그림 8은 S-MAC, T-MAC, IS-MAC 프로토콜에서 1 개의 데이터 패킷을 여러 홉 동안 전송하기 위해 소비되는 에너지를 나타낸다. 수식을 도출하기 위한 파라미터 값은 표 1과 같다<sup>[21],[7],[8]</sup>.

그림 8에서 크기가 100 byte인 데이터가 10 홉만큼 전송될 때 IS-MAC 프로토콜은 S-MAC 프로토콜보다 52%의 에너지 향상을 보이고 있으며, T-MAC 프로토콜보다 32%의 에너지 향상을 보인다. IS-MAC 프로토콜은 T-MAC 프로토콜과 같이 타이머를 사용하여 active 시간을 효율적으로 활용하고 있으며, 또한 다른 MAC 프로토콜보다 적은 수의 제어 패킷을 사용하고 있기 때문에 에너지 소비를 절감할 수 있다. 그림 9는 채널 사용률과 FRTS로 예약한 홉 수들의 관계를 보여준다. 채널

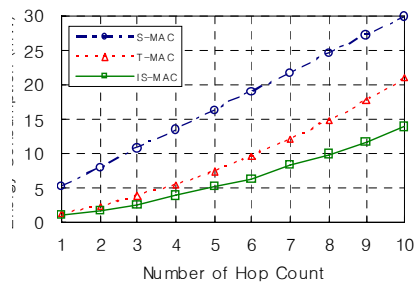


그림 8. 에너지 소비

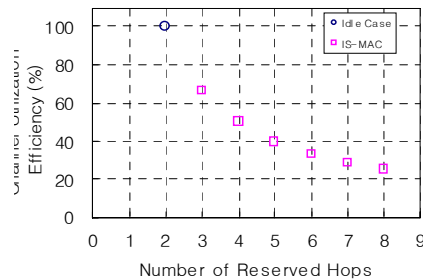


그림 9. 채널 효율성

표 1. 파라미터 값

프레임 길이	1 s
Active 시간	200 ms
Transmission rate	115 kbps
데이터 패킷 크기	100 byte
$TO_{T-mac}$	15 ms
$TO_{IS-mac}$	7.7 ms
$t_c$	1.7 ms
$E_{tx}$	30 mW/s
$E_{rx}$	12 mW/s
$E_{dl}$	12 mW/s

사용률은 S-MAC과 T-MAC 프로토콜에서 사용하고 있는 것과 같이 1 개의 RTS/CTS 패킷으로 예약 가능한 홉 수인 2 홉을 기준으로 계산하였다. 그림 9에서 알 수 있듯이 데이터 패킷을 전송할 수 있는 홉 수가 증가할수록 에너지 효율성은 향상되지만 채널 사용률이 떨어지기 때문에 에너지 효율성과 채널 사용률 간의 trade off 관계를 고려해야 한다.

위에서 살펴본 것처럼 IS-MAC 프로토콜은 FRTS 패킷으로 채널을 예약하기 때문에 S-MAC과 T-MAC 프로토콜보다 에너지 효율성이 더 뛰어나다. 또한 각 노드는 버퍼의 임계 값을 사용하여 노드별 공평성을 제공할뿐 아니라 앞장에서 계산한 에너지 절감보다 더 효율적으로 에너지 소비를 향상시킬 수 있다. 그 이유는 버퍼의 데이터 양이 적은 환경( $Q_{size} < Th_i$ )에서 노드는 데이터가 임계 값을 초과할 때까지 패킷을 전송하지 않기 때문이다. 그러나 IS-MAC 프로토콜과 같이 여러 홉을 예약하는 프로토콜에서는 채널을 예약하지 않는 프로토콜에 비하여 채널 사용률이 떨어지는 특성이 있기 때문에 FRTS 패킷으로 예약 가능한 홉 수를 적절하게 제한해 주어야 한다.

### V. 결론

본 논문은 센서 네트워크에서 idle listening 감소를 위해 타이머를 사용한 IS-MAC을 제안하였다. 또한 에너지 효율과 노드별 공평성을 향상시키기 위하여 임계 값을 적용한 버퍼를 소개하였다. IS-MAC 프로토콜에서는 채널 사용률을 향상시키기 위하여 1 개의 RTS/CTS 패킷으로 3 홉까지 데이터를 연속적으로 전송할 수 있게 하였다. 그리하여 노드별 공평성, 채널 용량 활용, 지연과 같은 QoS 뿐만 아니라 센서 네트워크 MAC 프로토콜의 중요

한 목적인 에너지 효율을 향상시켰다. 그리고 수학적 분석을 통해 IS-MAC 프로토콜이 기존의 센서 네트워크 MAC 프로토콜보다 에너지 효율성이 뛰어나다는 것을 확인하였다. 에너지 효율성 면에서 크기가 100 byte인 데이터가 10 홉만큼 전송될 때 IS-MAC 프로토콜은 S-MAC 보다 52%, T-MAC 프로토콜보다 32%의 향상을 보였다.

본 논문의 결과는 센서 네트워크의 에너지 향상과 노드별 공평성을 위하여 좋은 지침이 될 것임을 기대한다.

### 참고 문헌

- [1] W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control with coordinated Adaptive Sleeping for Wireless Sensor Networks", USC/ISI Technical Report ISI-TR-567, Jan. 2003.
- [2] T. V. Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wire-less Sensor Networks", in Proc. SenSys'03, Nov. 2003.
- [3] M. Stemm and R. H. Katz, "Measuring and reducing energy consumption of network interfaces in hand held devices", IEICE Transaction on Communications, Vol.E80-B, no.8, pp.1125-1131, Aug. 1997.
- [4] LAN MAN Standards Committee of the IEEE computer society: IEEE Std. 802.11-1999, Wireless LAN Medium Access Control (MAC) and Physical layer (PHY) specification IEEE 1999.
- [5] J. H. Kim, H. N. Kim, S. G. Kim, S. J. Choi, and J. Y. Lee, "Advanced MAC Protocol with Energy-Efficiency for Wireless Sensor Networks", Lecture Notes in Computer Science (LNCS), Springer, vol.3391, pp. 283-292, Jan. 2005.
- [6] C. Guo, L. Zhong, and J. Rabaey, "Low-Power Distributed MAC for Ad Hoc Sensor Radio Networks", in Proc. GLOBECOM '01, vol.5, pp.2944-2948, Nov. 2001.
- [7] RFM. TR1001 868.35 MHz Hybrid Transceiver.
- [8] <http://www.st.ewi.tudelft.nl/~koen/papers/mac-comp.pdf>



이 주 아 (Ju-a Lee)

정회원



2005년 2월 아주대학교 전자공학부 학사  
2005년 3월~현재 아주대학교 대학원 전자공학과 석사 과정  
<관심분야> 무선 LAN 보안, 센서 네트워크

민 승 옥 (Seung-wook Min)

정회원



1987년 2월 서울대학교 제어계측공학과 졸업  
1990년 2월 KAIST 전기및전자공학과 석사  
1999년 6월 Polytechnic Univ. 전자공학과 박사  
2004년 한국전자통신연구원

선임연구원

<관심분야> WLAN system, OFDM, CDMA, capacity analysis, propagation modeling

김 재 현 (Jae-hyun Kim)

정회원



1991년 2월 한양대학교 전산과 학사  
1993년 2월 한양대학교 전산과 석사  
1996년 8월 한양대학교 전산과 박사  
1997년 7월~1998년 6월 미국

UCLA 전기전자과 박사후 연수

1998년 11월~2003년 2월 Bell Labs, Performance Modeling and QoS Management Group, 연구원

2003년 3월~현재 아주대학교 전자공학부 조교수

<관심분야> 무선 인터넷 QoS, MAC 프로토콜, IEEE 802.11/15/16/20